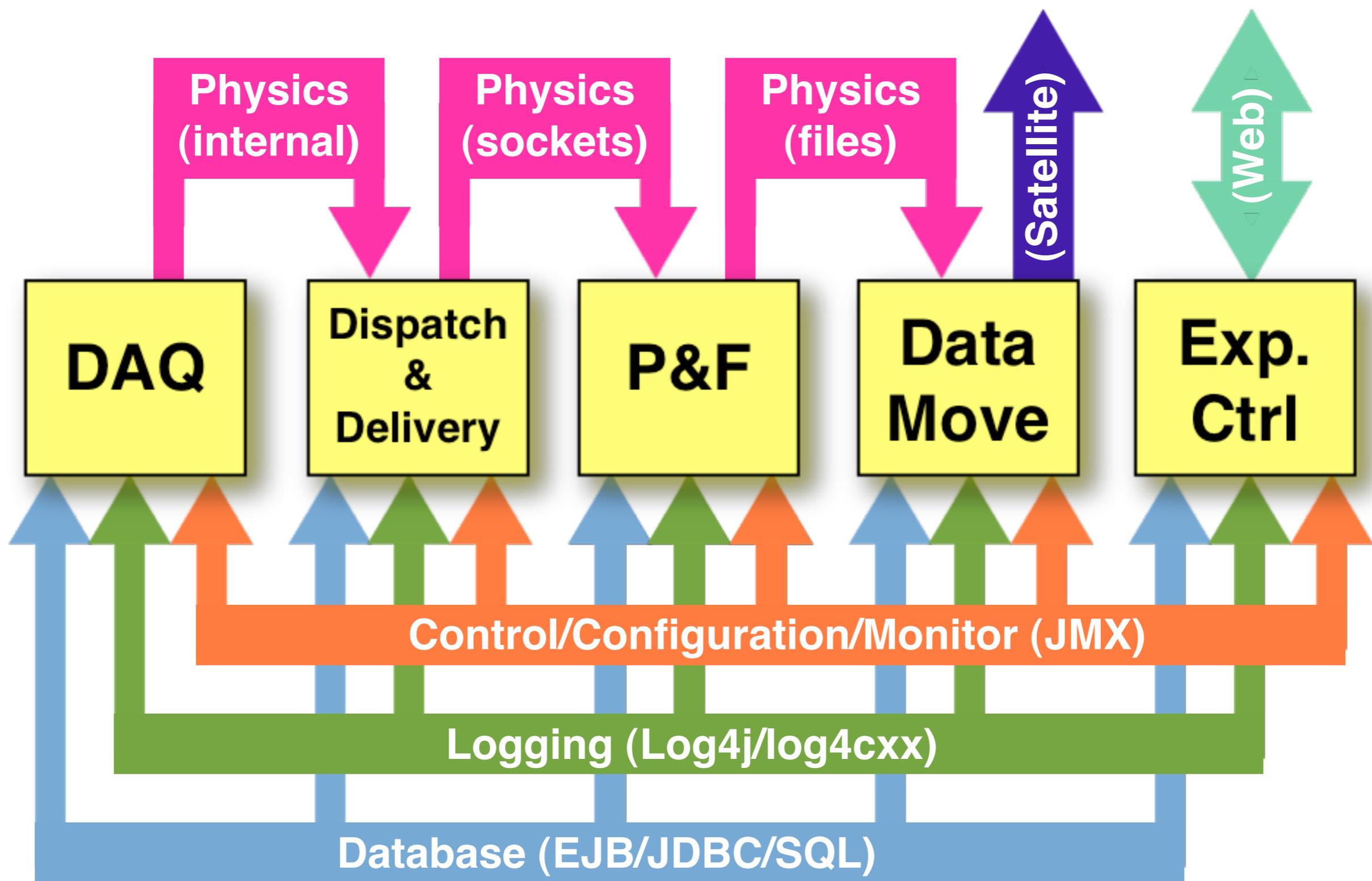


Experiment Control

Simon Patton



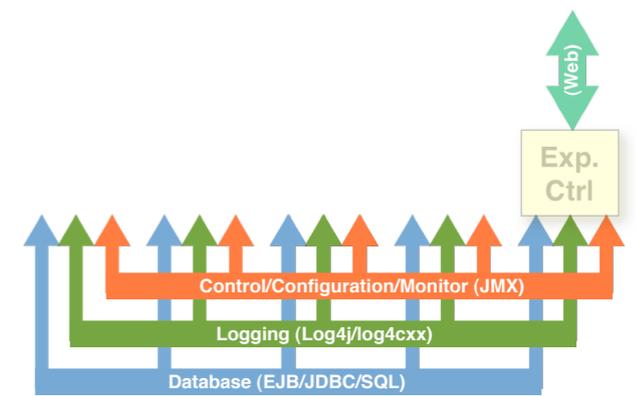
Demarcation

- **Logging**
 - ⊙ Manage messages pushed out by the systems.
- **Monitoring**
 - ⊙ Access to and storage of read only quantities.
- **Configuration**
 - ⊙ Access to and storage of read-write quantities.
- **Control**
 - ⊙ Management of execution.

Infrastructure

- JMX (Java Management Extensions) used as management interface.
- EJB (Enterprise Java Beans) and Java DataBase Connector (JDBC) used to make information persistent.
- JSP (Java Server Pages) used to prepare dynamic web pages.
- Portal used to manage users and allow customization of web pages.

JBoss



- Off-the-shelf J2EE application server.
- Built around a Java Management Extensions (JMX) backbone.
- Provides Enterprise Java Beans (EJB) and Java Server Pages (JSP) containers.
- Provides **jmx-console** application that allows access to Managed Beans (MBeans).

jmx-console



JMX Agent View

wallaby.dhcp.lbnl.us

ObjectName Filter (e.g. "jboss:*", "*:service=invoker,*"):

Apply Filter

Clear Filter

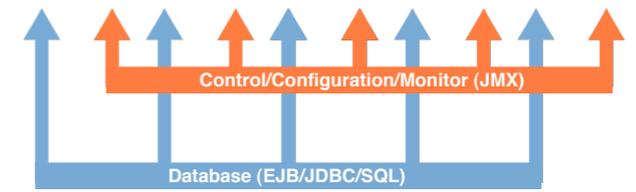


[icecube.daq.dispatch](#)

- [acme-aspect=configuration.component=idd.dispatch=DevNull.idd=dispatch](#)
- [acme-aspect=configuration.component=idd.idd=producer.producer=Random](#)
- [acme-aspect=control.component=idd.dispatch=DevNull.idd=dispatch](#)
- [acme-aspect=control.component=idd.idd=producer.producer=Random](#)
- [acme-aspect=monitor.component=idd.dispatch=DevNull.idd=dispatch](#)
- [acme-aspect=monitor.component=idd.idd=producer.producer=Random](#)

JBoss Support

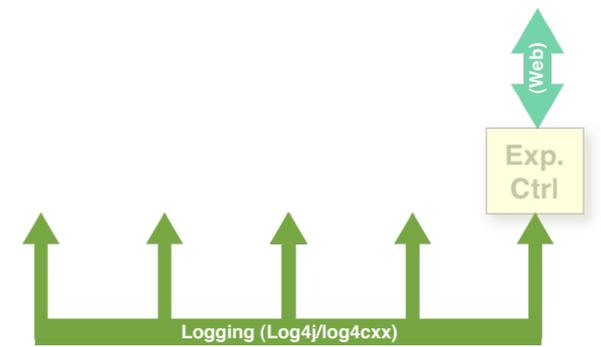
(Chris Day/Simon Patton/Martin Stoufer/
Keith Beattie)



- Define installation and deployment protocols.
- Supply classes for common IceCube specific tasks.
- Educate developers.

Logging

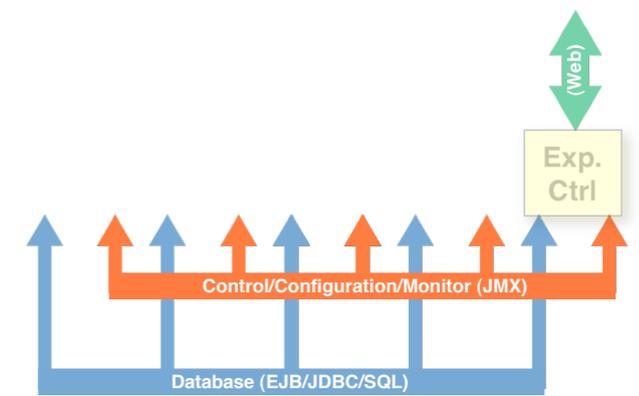
(Akbar Mokhtarani/Chris Day)



- Define logging interface to be used.
- Central “Concentrator” for all **WARN**, **ERROR** and **FATAL** messages.
- Compatibility with Offline **C++** logging package.
- Presentation GUI to access messages.

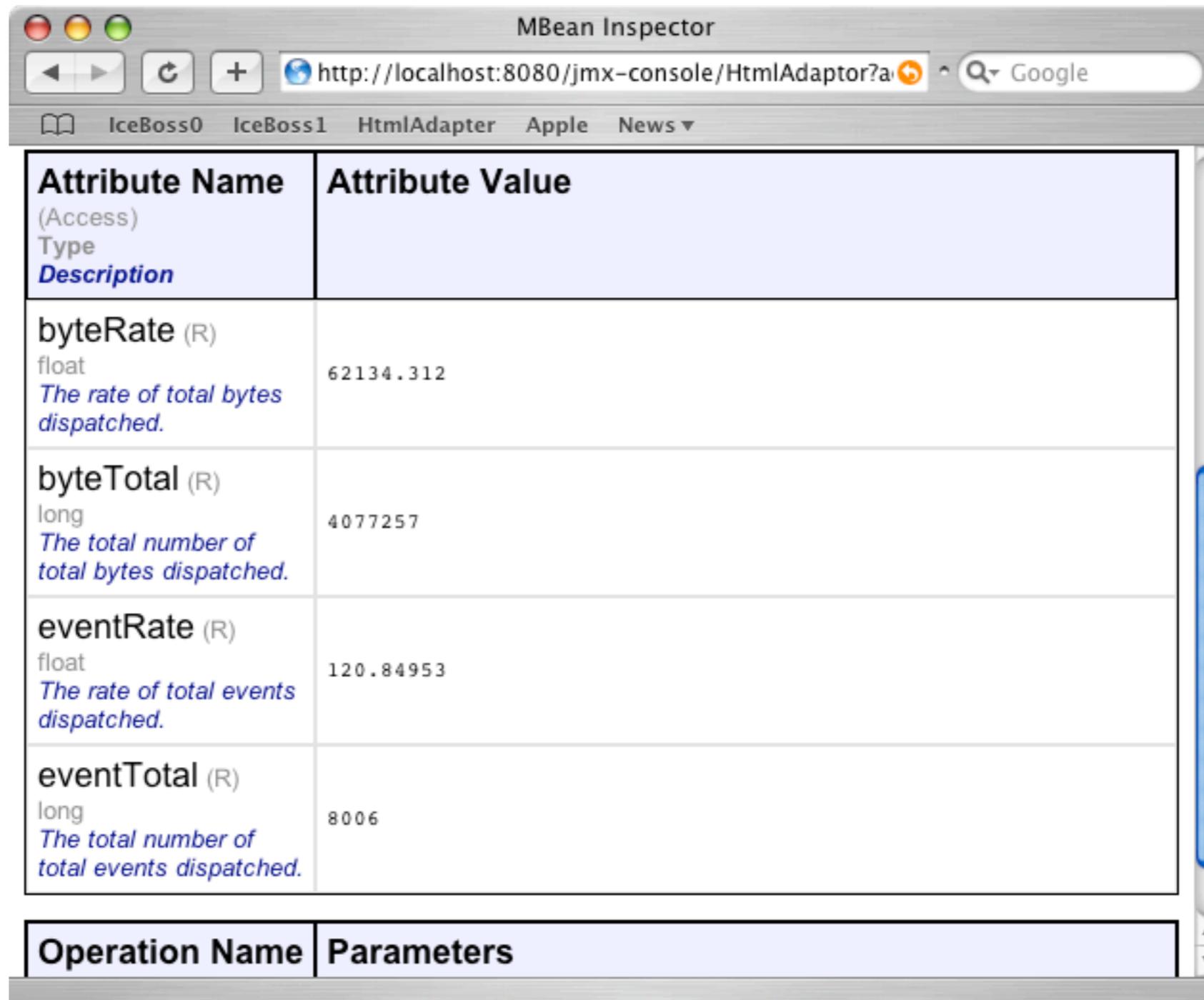
Monitoring

(Simon Patton/Akbar Mokhtarani)



- Saving of attributes using JMX infrastructure and JDBC.
- Supply of “Standard” monitoring classes.
- Display built using Portal and Java Server Pages (JSPs).
- Published via “acme-aspect=monitor”.

Monitoring Example



The screenshot shows a web browser window titled "MBean Inspector" with the URL <http://localhost:8080/jmx-console/HtmlAdaptor?a>. The browser tabs include "IceBoss0", "IceBoss1", "HtmlAdaptor", "Apple", and "News". The main content area displays a table of MBean attributes for the "IceBoss" component.

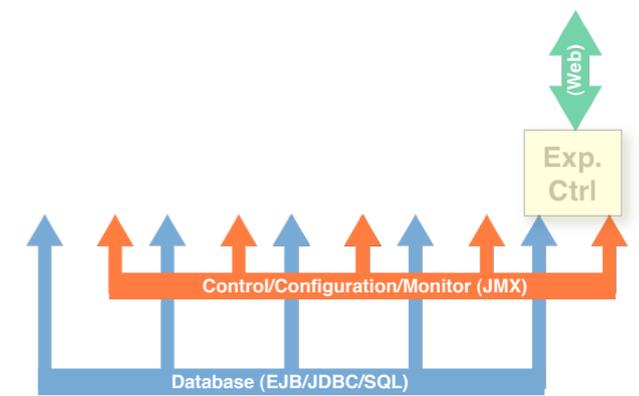
Attribute Name (Access) Type <i>Description</i>	Attribute Value
byteRate (R) float <i>The rate of total bytes dispatched.</i>	62134.312
byteTotal (R) long <i>The total number of total bytes dispatched.</i>	4077257
eventRate (R) float <i>The rate of total events dispatched.</i>	120.84953
eventTotal (R) long <i>The total number of total events dispatched.</i>	8006

Below the attribute table, there is a section for operations with the following header:

Operation Name	Parameters
----------------	------------

Configuration

(Akbar Mokhtarani/Simon Patton/Chris Day)



- Internal saving and setting of attributes using EJBs.
- External saving and setting of attributes using JMX infrastructure and JDBC.
- Interface built using Java Server Pages (JSPs).
- Published via “acme-aspect=configuration”.

Configuration Example

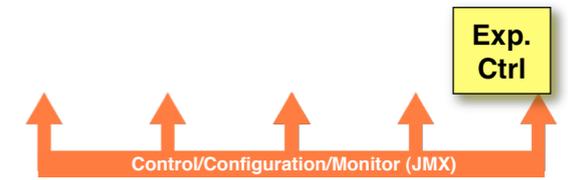
The screenshot shows the MBean Inspector web console. The browser address bar displays `http://localhost:8080/jmx-console/HtmlAdaptor?a`. The breadcrumb navigation shows `IceBoss0 > IceBoss1 > HtmlAdaptor > Apple > News`. The main content area displays the configuration for the `dispatchAndService` MBean. The configuration is as follows:

Property Name	Value
<code>dispatchAndService</code> (RW) javax.management.ObjectName <i>The service that binds the Dispatcher into JNDI.</i>	<code>icecube.daq.dispatch:acme-aspect=configuration,co</code> View MBean
<code>eventSize</code> (RW) int <i>The mean size, in bytes, of the events produced.</i>	512
<code>eventVariation</code> (RW) float <i>The spread, as a fraction of event size, of event sizes.</i>	0.5
<code>eventsPerDispatch</code> (RW) int <i>The mean number of events per call to the Dispatcher.</i>	16
<code>targetByteRate</code> (RW) int <i>The mean rate, in bytes/s, with which to produce data.</i>	524288

At the bottom of the configuration area, there is an `Apply Changes` button.

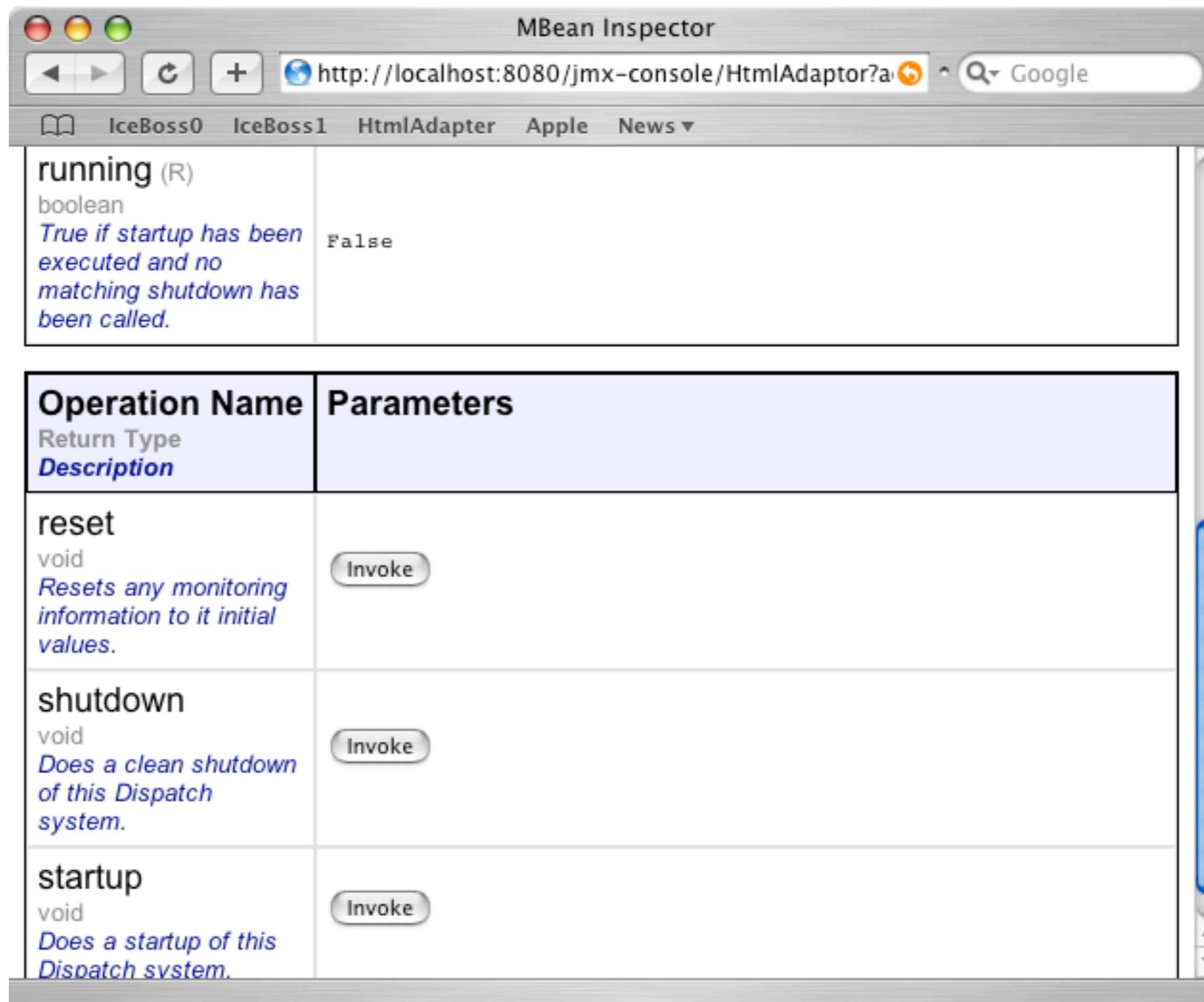
Control

(Chris Day)



- Coordinate inter-system information.
- Implement nested state machines.
- Published via “acme-aspect=control” MBeans.

Control Example



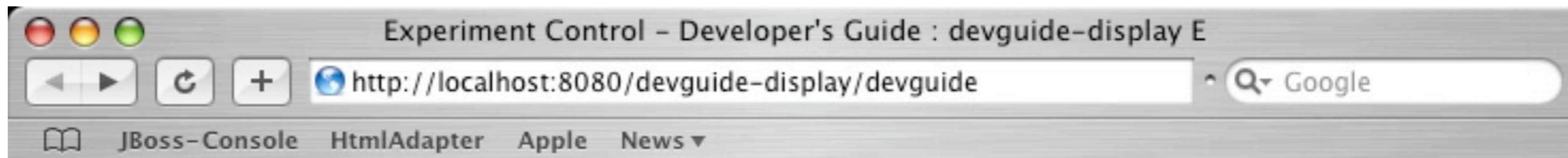
The screenshot shows a web browser window titled "MBean Inspector" with the URL `http://localhost:8080/jmx-console/HtmlAdaptor?a`. The browser tabs include "IceBoss0", "IceBoss1", "HtmlAdaptor", "Apple", and "News".

The main content area displays the following information:

running (R)
boolean
True if startup has been executed and no matching shutdown has been called. `False`

Operation Name	Parameters
reset void <i>Resets any monitoring information to it initial values.</i>	<input type="button" value="Invoke"/>
shutdown void <i>Does a clean shutdown of this Dispatch system.</i>	<input type="button" value="Invoke"/>
startup void <i>Does a startup of this Dispatch svstem.</i>	<input type="button" value="Invoke"/>

Presentation Example



Example devguide-display E

Execute the Application

Run

Status of Splicer

Number of Object seen by Splicer:

Update

Status of Generators

Generator ID	interval/ms
0	23
1	96
2	75

Status and Plans

- Most infrastructure installed.
- In use at the Pole for all systems except TestDAQ.
 - ⊙ Each system has its own `jmx-console`.
- In the coming year concentrating on:
 - ⊙ inter-system coordination logic;
 - ⊙ support for seamless experiment expansion;
 - ⊙ presentation layer for all systems.



“That's all Folks!”